

REMARKS

Reconsideration of this application, as amended, is respectfully requested.

Claims 1-27 are pending. Claims 1-27 stand rejected.

Claims 1, 10, and 19 have been amended. No claims have been cancelled. No claims have been added. Support for the amendments is found in the specification, the drawings, and in the claims as originally filed. Applicants submit that the amendments do not add new matter.

Rejections Under 35 U.S.C. § 103(a)

Claims 1, 2, 4-6, 9-11, 13-15, 18-20, and 22-24 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over U.S. Patent Application No. 2003/0079210 to Markstein et al. (“Markstein”) in view of “An Experimental Study of Several Cooperative Register Allocation and Instruction Scheduling Strategies”, International Symposium on Microarchitecture, Proceedings of the 28th annual international symposium on Microarchitecture, pages 169-179, 1995 of Pollock et al. (“Pollock”).

Applicants have amended claim 1 to particularly point out that modifying the program is performed to use an expanded register set for a routine in the program transparently to execution of the routine.

Amended claim 1 reads as follows:

A machine-implemented method comprising:
analyzing one or more instructions of a program; and
modifying the program to use an expanded register set for a routine in the program transparently to execution of the routine.

(Amended claim 1) (emphasis added)

Markstein discloses allocation of registers for an intermediate code of a source program in a compiler. Markstein discloses saving and restoring registers, which store operands (“instructions”) for the source program. Markstein discloses

For example, when the number of available real registers is insufficient to store all of the intermediate values in the intermediate registers that are specified by the code instructions, some intermediate values may have to be stored in other memory. The process of temporarily storing data from a real register to another memory location is referred to as spilling. Generally, spilling involves performing a store operation, followed by one or more reload operations. A spill operation causes data contained in a real register to be stored in another memory location, such as a runtime stack. Each reload operation causes the data to be loaded or copied from the other memory location into a real register. Reload operations are performed when the data is required for a calculation. A prologue and an epilog may be used to save and restore callee-saved registers (e.g., registers storing operands preserved for an extended period of time during execution of the translated code). A prologue and epilog typically includes code executed before and after a subroutine or program. For example, when a prologue is executed stack space may be allocated for saving necessary context, such as saving callee-saved registers. When an epilog is executed, the compiler may restore any necessary registers.

(Markstein, [0006]) (emphasis added)

In particular, Markstein discloses

In step 220, optimization, such as redundant code elimination and constant propagation are performed for translated intermediate language instructions. In step 222, the intermediate language instructions are converted into target instructions. Additional register allocation may be needed if a single intermediate level instruction expands into more than one target level instruction.

(Markstein, [0051]) (emphasis added)

Thus, Markstein, in contrast, discloses allocating additional registers to store an expanded instruction (“expanded routine”) of the program. Markstein, in contrast, discloses the additional registers, which are not transparent to execution of the routine of the program, because they store the expanded routine of the program.

The Examiner noted that Markstein fails to disclose a limitation of claim 1 of modifying the program to expand a register set of a routine in the program. Additionally, Markstein fails to

disclose modifying the program to use an expanded register set for a routine in the program transparently to execution of the routine, as recited in amended claim 1.

Pollock discloses cooperation between the instruction scheduling and register allocation in optimizing compilers.

More specifically, Pollock discloses

A scheduler that rearranges code within a basic block in isolation of the rest of the program is called a local scheduler; a scheduler that moves instructions across basic blocks by considering the effects of code movement on a global level is called a global scheduler. The goal of an ambitious register allocator is to allocate the machine's physical registers to program values to minimize the number of run-time memory access. Register allocation techniques are either local, global, or interprocedural depending on whether the allocator attempts an assignment of registers to values within basic blocks (chain) in isolation of other basic blocks, across basic blocks of a procedure, or across procedure boundaries, respectively.

(Pollock, Introduction, p. 169) (emphasis added)

Thus, Pollock, in contrast, discloses instruction scheduling that rearranges code within the block of the program and register allocating techniques that assign registers for the blocks ("routine") of the program, i.e. not transparent to the execution of the routine of the program.

Pollock, similar to Markstein, fails to disclose, teach, or suggest modifying the program to use an expanded register set for a routine in the program transparently to execution of the routine, as recited in amended claim 1.

As discussed above, it is respectfully submitted that Markstein does not teach or suggest a combination with Pollock, and that Pollock does not teach or suggest a combination with Markstein. It would be impermissible hindsight, based on applicants' own disclosure, to combine Markstein and Pollock. Markstein discloses allocating physical registers for intermediate code generated between a machine-readable code and a source code. Pollock, in contrast, discloses communication and cooperation between register allocation and global and local instruction scheduling.

Furthermore, even if Markstein and Pollock were combined, such a combination would lack the limitation of amended claim 1 of modifying the program to use an expanded register set for a routine in the program transparently to execution of the routine.

Therefore, applicants respectfully submit that amended claim 1 is not obvious under 35 U.S.C. § 103 (a) over Markstein in view of Pollock.

Because amended independent claims 10 and 19 contain at least the same limitations as amended claim 1, applicants respectfully submit that amended claims 10 and 19 are likewise not obvious under 35 U.S.C. § 103 (a) over Markstein in view of Pollock.

Given that claims 2-6 and 8-12 depend, directly or indirectly, from amended independent claims 1 and 7 respectively, and add additional limitations, applicant respectfully submits that claims 2-6 and 8-12 are likewise not obvious under § 103 (a) over Markstein in view of Pollock.

Given that claims 2-9, 11-18, and 20-27 depend, directly or indirectly, from amended claims 1, 10, and 19 respectively, and add additional limitations, applicants respectfully submit that claims 2-9, 11-18, and 20-27 are likewise not obvious under 35 U.S.C. § 103 (a) over Markstein in view of Pollock.

It is respectfully submitted that none of the references cited by the Examiner disclose, teach, or suggest such limitation of amended claim 1.

Claims 3, 12 and 21 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over Markstein in view of Pollock, and further in view of U.S. Patent No. 5,644,709 to Austin (“Austin”).

Austin, in contrast, discloses a method for detecting memory access errors, which occur while executing a computer program. Austin, similarly to Markstein and Pollock does not disclose, teach, or suggest the limitation of amended claim 1 of modifying the program to use an expanded register set for a routine in the program transparently to execution of the routine.

Consequently, even if Austin, Markstein, and Pollock were combined, such a combination would lack such limitation of amended claim 1.

Therefore, applicants respectfully submit that amended claim 1 is not obvious under 35 U.S.C. § 103 (a) over Markstein in view of Pollock, and further in view of Austin.

Because amended independent claims 10 and 19 contain at least the same limitations as amended claim 1, applicants respectfully submit that amended claims 10 and 19 are likewise not obvious under 35 U.S.C. § 103 (a) over Markstein in view of Pollock and further in view of Austin.

Because claims 3, 12, and 21 depend, directly or indirectly, from amended independent claims 1, 10, and 19 respectively, and add additional limitations, applicants respectfully submit that claims 3, 12, and 21 are likewise not obvious under 35 U.S.C. § 103 (a) over Markstein in view of Pollock, and further in view of Austin.

Claims 7, 8, 16, 17, and 25-27 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over Markstein in view of Pollock, and further in view of U.S. Patent No. 6,256,777 to Ackerman (“Ackerman”).

Ackerman, in contrast, discloses debugging a program using a debug information file and hidden breakpoints inserted into the machine code. Ackerman, similarly to Markstein and Pollock fails to disclose the limitation of amended claim 1 of modifying the program to use an expanded register set for a routine in the program transparently to execution of the routine

Consequently, even if Austin, Markstein, Pollock, and Ackerman were combined, such a combination would lack such limitation of amended claim 1.

Therefore, applicants respectfully submit that amended claim 1 is not obvious under 35 U.S.C. § 103 (a) over Markstein in view of Pollock, and further in view of Ackerman.

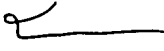
Because amended independent claims 10 and 19 contain at least the same limitations as amended claim 1, applicants respectfully submit that amended claims 10 and 19 are likewise not obvious under 35 U.S.C. § 103 (a) over Markstein in view of Pollock, and further view of Ackerman.

Because claims 7, 8, 16, 17, and 25-27 depend, directly or indirectly, from amended independent claims 1, 10, and 19 respectively, and add additional limitations, applicants respectfully submit that claims 7, 8, 16, 17, and 25-27 are likewise not obvious under 35 U.S.C. § 103 (a) over Markstein in view of Pollock, and further in view of Ackerman.

It is respectfully submitted that in view of the amendments and arguments set forth herein, the applicable rejections and objections have been overcome. If there are any additional charges, please charge Deposit Account No. 02-2666 for any fee deficiency that may be due.

Respectfully submitted,

BLAKELY, SOKOLOFF, TAYLOR & ZAFMAN LLP

Date: 3/9/05 By: 
Michael J. Mallie
Reg. No. 36,591

12400 Wilshire Boulevard
Seventh Floor
Los Angeles, California 90025
(408) 720-8300